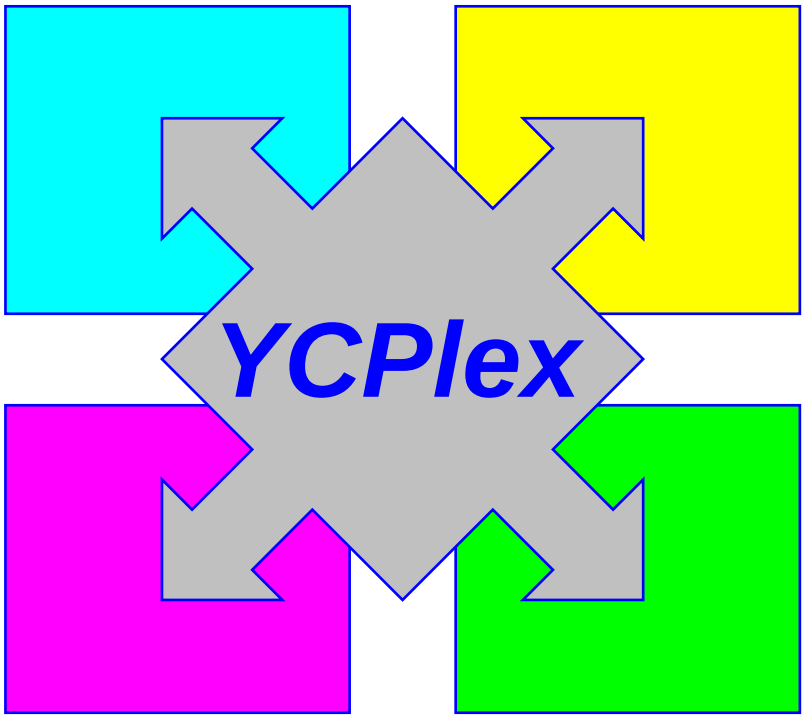


YCPlex Version 2, Release 0
Mod 0

U s e r ' s G u i d e

YCOS Yves Colliard Software GmbH
Fremersbergstr. 45
D-76530 Baden-Baden

Tel: (D) 07221/9708384
Fax: (D) 0322 2374 2352

e-Mail: ycos@ycos.de
Home: http://www.ycos.de

Copyright YCOS Yves Colliard Software GmbH

Copyright YCOS Yves Colliard Software GmbH 2009-10

All rights reserved. Duplication or disclosure only with explicit approval of
YCOS Yves Colliard Software GmbH.

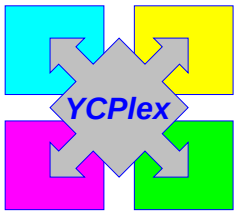


Table of contents

1 YCPlex – YCOS Sysplex 5
1.1 YCPlex Overview 5
1.2 YCPlex processing 10
1.2.1 Send Requests – SEND 10
1.2.2 Receiving the request – SEND 12
1.2.3 Receiving the response – RESP 14
1.3 YCPlex samples 16
1.3.1 System Rexx sample – YCXCFSND 16
1.3.2 Sample – YCPLEXQY 20
1.3.3 Assembler sample – YCXCFASM 21
2 Installation 22
2.1 STC: Started Task 22
2.2 RACF 22
2.3 Performance 23
3 Operation 24
3.1 Start of YCPlex 24
3.2 Stop of YCPlex 24
3.3 Modules, Versionen, PTF and Compile of YCPlex 24
3.4 YCXCFREX System Rexx Diagnosis 25
3.5 Messages of YCPlex 26
3.5.1 Messages of YCXCFCMD: 27
3.5.2 Messages of YCXCFTIM: 28
3.5.3 Messages of YCXCFSTC: 29
3.5.4 Messages of YCXCFWRK: 30
3.5.5 Messages of YCXCFMGO: 31
3.5.6 Messages of YCXCFMGI: 32
3.5.7 Messages of REXX – YCXCFREX: 33
3.5.8 Messages of REXX – YCXCFSND: 34
3.5.9 Messages von Assembler – YCXCFASM: 35
4 YCPlex Support 36
5 YCPlex Version and Release 37
5.1 Version 2 Release 0 and PTFs 37
5.2 Version 1 Release 2 and PTFs 37
5.3 Version 1 Release 1 and PTFs 38
5.4 Version 1 Release 0 und PTFs 40
6 Index 41

Copyright YCOS Yves Colliard Software GmbH

Table of figures

Figure 1: YCPlex overview 5
Figure 2: Send Requests 10
Figure 3: receiving the request 12
Figure 4: receiving the response 14
Figure 5: sample system rexx – request – YCXCFSND 19
Figure 6: sample system rexx – response – YCXCFMRG 19
Figure 7: sample system rexx – response – YCXCFTST 20
Figure 8: YCPlex started task 22

Copyright YCOS Yves Colliard Software GmbH

1 YCPlex – YCOS Sysplex

1.1 YCPlex Overview

YCPlex – YCOS sysplex communication – provides a flexible interface within a sysplex; the standard XCF *(cross system coupling facility – called basic sysplex)* services are used. The YCPlex interface can be used to send tasks/commands/actions – SEND – to other systems; the answer(s) can be – RESP – returned to the caller and queries – QURY – are also provided.
Tasks/command/actions can be sent to ALL systems *(*ALL)*, to ALL others systems *(*OTHER)* or to a selected system *(system name)*.
The response have to be sent to a dedicated/selected system.

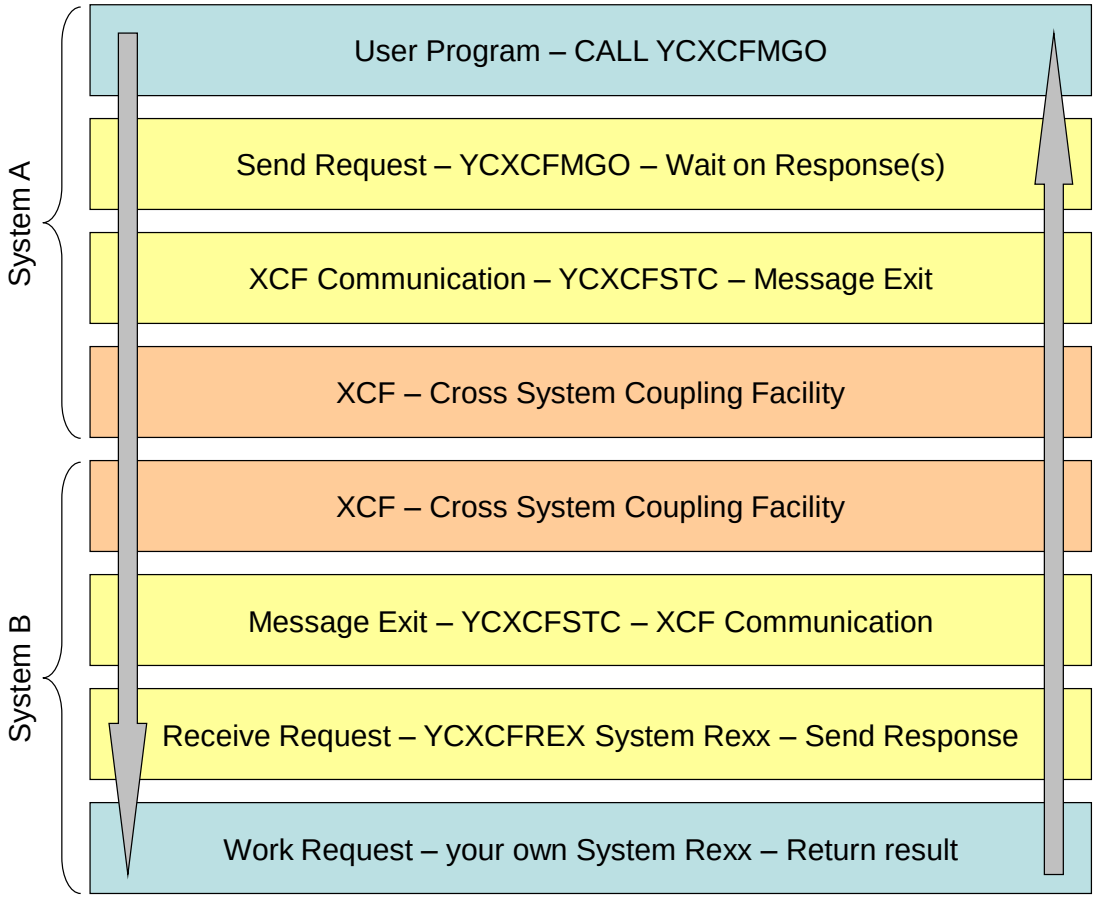
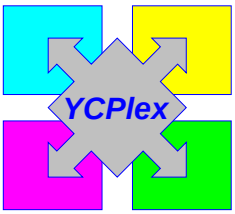

System A
System B
User Program – CALL YCXCFMGO
Send Request – YCXCFMGO – Wait on Response(s)
XCF Communication – YCXCFSTC – Message Exit
XCF – Cross System Coupling Facility
XCF – Cross System Coupling Facility
Message Exit – YCXCFSTC – XCF Communication
Receive Request – YCXCFREX System Rexx – Send Response
Work Request – your own System Rexx – Return result

Figure 1: YCPlex overview

Copyright YCOS Yves Colliard Software GmbH



YCPlex needs a startet task and uses the system rexx interface; it will also create/join an own XCF Group – YCXCFGRP; the member name will be the system name . One started task per system will be needed.

The **YCPlex** message interface – YCXCFMGO – must be authorized; any address space type can use the interface.

Following components/programms are part of YCPlex:

- **YCXCFSTC** : the started task of YCPlex is responsible for the following tasks:
  - YCPlex can only be started once per system – this will be checked; YCPlex uses NTC *(Name Token Callable Services)* with a token name „YCXCFSTC+System-Name“ to check it
  - It loads the message exit – YCXCFMGI – in the storage. This task *(exit)* becomes control as an message is coming in *(received by XCF)*
  - Up to 100 messages can be queued in the storage
  - The XCF-group YCSCFGRP with member name “system name” will be created/joined
  - A NTC „YCXCFSTC+System-Name „YCXCFSTC+System-Name“ will be create to keep some internal information: member-token *(for YCXCFMGO)*, getmain-area-pointer *(Debug)* and ECB-pointer *(Debug)*
  - the command-subtask will be attached – YCXCFCMD
  - an abend-522-elimination-subtask will attached – YCXCFTIM – no need for TIME=1440 since no long wait time will occurred!
  - the work-message-subtask will be attached – YCXCFWRK. This task will be posted by YCXCFMGI to process the received message
  - the started task wait on the ECBs *(Command, Time, Work)*; normaly until a Stop command is entered
  - after the Stop command is given, the subtasks will be detach and the storage will be freed
- **YCXCFCMD**: the subtask of YCXCFSTC will process modify and stop commands. Following commands are supported:
  - STOP/P, MODIFY/F SHUTDOWN/STOP/P: YCPlex will be stopped
  - MODIFY/F LOADMODS: all YCPLex started task modules will be displayed; also their current version and PTF levels
  - Unknown commands will produce an error message
- **YCXCFTIM**: the suubtask YCXCFTIM is designed to prevent abends S522
- **YCXCFWRK**: this subtask will be posted on every “SEND” request; the information/request will be forwared to system rexx – YCXCFREX
- **YCXCFMGI**: this SRB exit routine will be loaded in storage by YCXCFSTC and will be defined as the message-exit routine of the XCF Group YCXCFGRP. The SRB routine becomes control on SEND requests and on responses; following tasks will be executed:
  - **SEND**: the requests will be serialized and put to the message stack; YCXCFWRK will then posted

Copyright YCOS Yves Colliard Software GmbH

- **RESP**: on responses the address space and name token of the SEND requestor *(created thru YCXCFMGO during SEND)* will be used, to copy the response in the defined area/address space *(cross memory)* and post the waiting address space

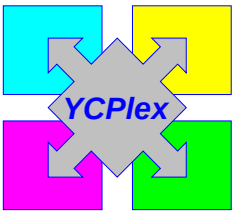
- **YCXCFMGO**: can be called from any address space, but must be authorized. YCXCFMGO is used for sending requests and responses. Responses can only be used as "Reply" to a request! The use of the YCXCFMGO interface is subject to a RACF protection. SEND requests are protected by following profiles in the class FACILITY:
  - YCPLEX.SEND.function.sysname

  READ permission is needed to send a SEND request to the selected system
  The RESP request is protected by following FACILITY profile:
  - YCPLEX.RESP

  Only the user of the YCPLEX started task should get READ permission
  Following parameters – macro MGOPARM- can be used:
  - **SEND**

```
PRM_L     DS    CL4     Length
REQUEST   DS    CL4     Request SEND
SYS_SEL   DS    CL8     System Name selection
*                       *ALL    - to alls systems
*                       *OTHER  - all other systems
*                       sysname - this system
PRM_NTC   DS    CL16    reserved
PRM_PTRR  DS    CL4     reserved
          ORG   PRM_PTRR
PRM_TIMO  DS    CL4     Time out
PRM_STCK  DS    CL8     reserved
PRM_PARM  DS    0CL1    Request – Max 896
```

- 2 Bytes - HH - Length of parms
- 8 Bytes – System Rexx receiving control
- up to 886 Bytes – request parameter sufficient space for the answers – Max 4+((896+8)*32) systems
- 4 Bytes - LLLL - Number of responses per Response
- 8 Bytes - SYSTEM - System name of response
- 2 Bytes - HH - Length of response
- if 0000 - Then no response
- up to 894 Bytes – Response
- if Response starts with YXCFR then Error message from YCXCFREX



- **RESP**
  - PRM_L DS CL4 Length
  - REQUEST DS CL4 Request RESP
  - SYS_SEL DS CL8 System Name selection
    * sysname - this system
  - PRM_NTC DS CL16 NTC Name – TTOKEN
    Filled during SEND
  - PRM_PTRR DS CL4 Pointer to Response Area
    Filled during SEND
  - PRM_PARM DS 0CL1 Response
  - 2 Bytes - HH - Length of Response
  - up to 894 Bytes – Response
- **QURY**
  - PRM_L DS CL4 Length
  - REQUEST DS CL4 Request QURY
  - SYS_SEL DS CL8 not used
  - PRM_NTC DS CL16 not used
  - PRM_PTRR DS CL4 not used
  - PRM_PARM DS 0CL1 Response
    - 4 Bytes - LLLL - Number of responses per Response
    - 8 Bytes - SYSTEM - System name of response
    - 2 Bytes - HH - Length of response
    - if 0000 - Then no response
    - up to 894 Bytes – Response

Copyright YCOS Yves Colliard Software GmbH

- **YCXCFREX**: this System-Rexx *(TSO environment and without time limit)* – must be installed in SYS1.SAXREXEC or the concatenation – gets control from YCXCFWRK when a SEND request arrives. YCXCFREX receives the information *(956 buffer)* in two variables *(AXREXX restriction!)*: YCXCFIN1 and YCXCFIN2 – these are reassembled in the variable YCXCFINF. The MSG_PREQ variable will contain the name of a user system REXX, that will then be given control with parameters

- Parameter 1 – MSG_DIAG
- Parameter 2 – YCXCFINF

  This information can be processed as follows in the user REXX *(see samples YCXCFRMG and YCXCFTST)*:

```
MSG_DIAG =substr(arg(1),1,32)         /* extract Diagnosis         */
MSG_LEN  =substr(arg(1),34,4)         /*         length            */
MSG_INFO =substr(arg(1),38,4)         /*         info: SEND        */
MSG_STCK =substr(arg(1),42,8)         /*         Store Clock       */
MSG_SYSN =substr(arg(1),50,8)         /*         sending System    */
MSG_USER =substr(arg(1),58,8)         /*         sending User Id   */
MSG_NTOK =substr(arg(1),66,16)        /*         sending Name Tok  */
MSG_PTRR =substr(arg(1),82,4)         /*         sending Data Prt  */
MSG_PARM =substr(arg(1),86)           /* Length and Data           */
MSG_PLGT =x2d(c2x(left(MSG_PARM,2))) /*   length of Data          */
MSG_PREQ =substr(arg(1),88,8)         /*   Requested Function      */
MSG_PDAT =substr(arg(1),96)           /*   Parameter Data          */
```

  The user system REXX can now perform the desired function and returns the answer in the stack – QUEUE – and a return code may also be delivered:

```
QUEUE response
EXIT [rc]
```

  YCXCFREX uses YCXCFMGO to send the result/reponse back tot he requestor – RESP.

1.2 YCPlex processing

The YCPlex processing consists of 4 phases and these are described in the next few pages.

1.2.1 Send Requests – SEND

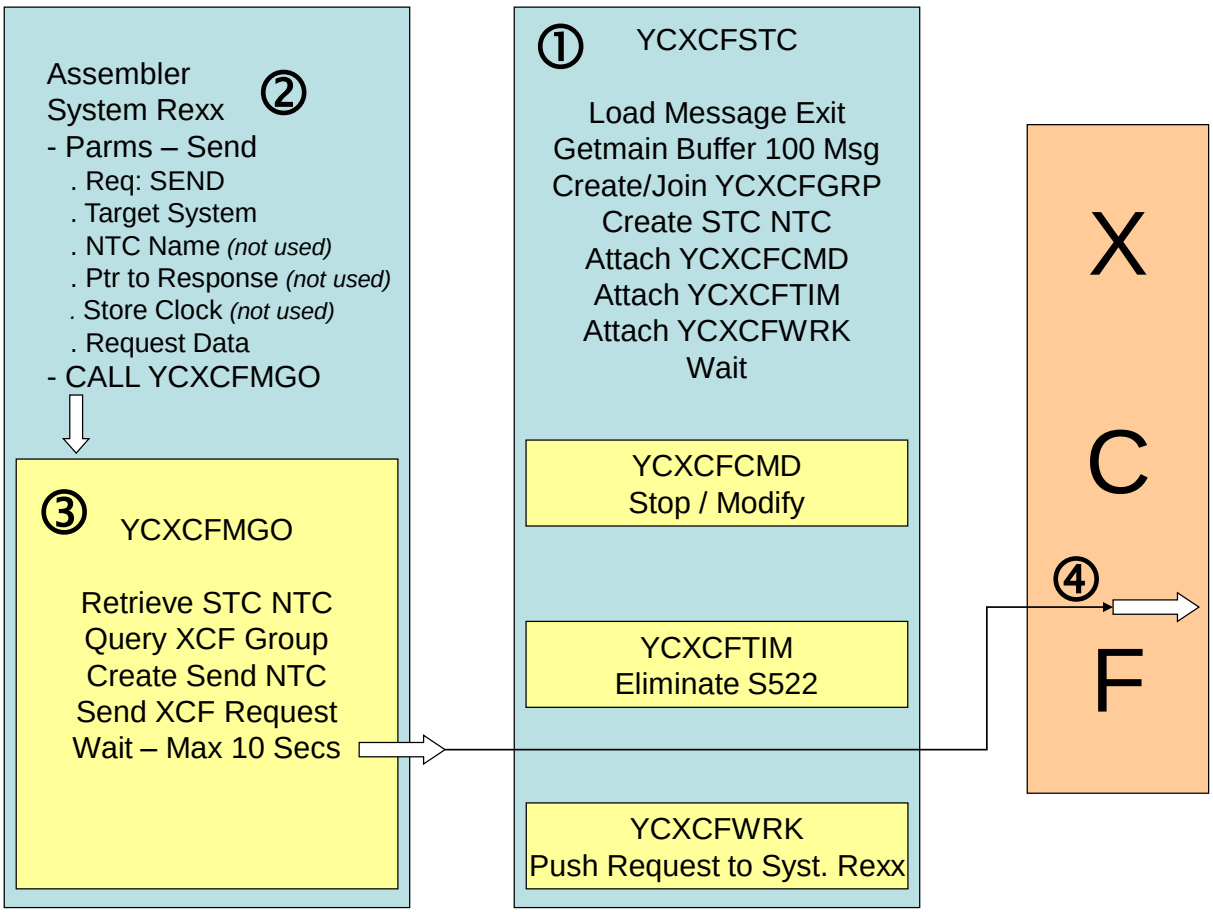


Figure 2: Send Requests

1. the started task *(this could be also a batch job)* YCXCFSTC has to be started; it is required to use YCPlex services. The started task has following functions:
   a. the message exit YCXCFMGI is loaded in storage
   b. a buffer able to hold 100 messages will be created
   c. the XCF group YCXCFGRP with a member named „Sysname“ will be created od joined
   d. a name-token is created – named YCXCFSTC+sysname – and is holding the member-token

Copyright YCOS Yves Colliard Software GmbH

e. 3 subtasks are initiated:
    i. YCXCFCMD: command Interface
    ii. YCXCFTIM: abend S522 elimination
    iii. YCXCFWRK: send request
f. the started task goes into a state

2. an authorized assembler program or a system REXX can be used to call the program YCXCFMGO which make a SEND request. Following parameters are required *(see MGOPARM Makro)*:
    a. request: SEND *(4 Bytes)*
    b. target system *(8 Bytes)*; following parameters are allowed:
        i. *ALL – the request will be sent to all members – including this system
        ii. *OTHER – the request will be sent to all "other" members
        iii. system name – the request will be sent to the selected member
    c. the request must pre prefixed by a length field *(2 Bytes – length including length field!)*. The request can *(including length field)* be 896 bytes long
3. the YCXCFMGO program receives control and performs the following actions:
    a. the NTC *(Name Token Callable Services)* of the started task will be read to get member token – this is required to send data over this XCF member
    b. the XCF group will be queried to find out all members within the sysplex
    c. a NTC for this SEND request will be created using the TCB-token as name *(Task Control Block)*
    d. the request ist send per XCF to the selected target(s)
    e. YCXCFMGO waits for a response or until a time out occurs; the default time out is defined to 10 secondes; the MSG_TIMO can be used to change the time out value up to 1440 seconds
4. XCF will send the request over CTC *(Channel To Channel)* or CF *(Coupling Facility)* to the selected member(s)

1.2.2 Receiving the request – SEND

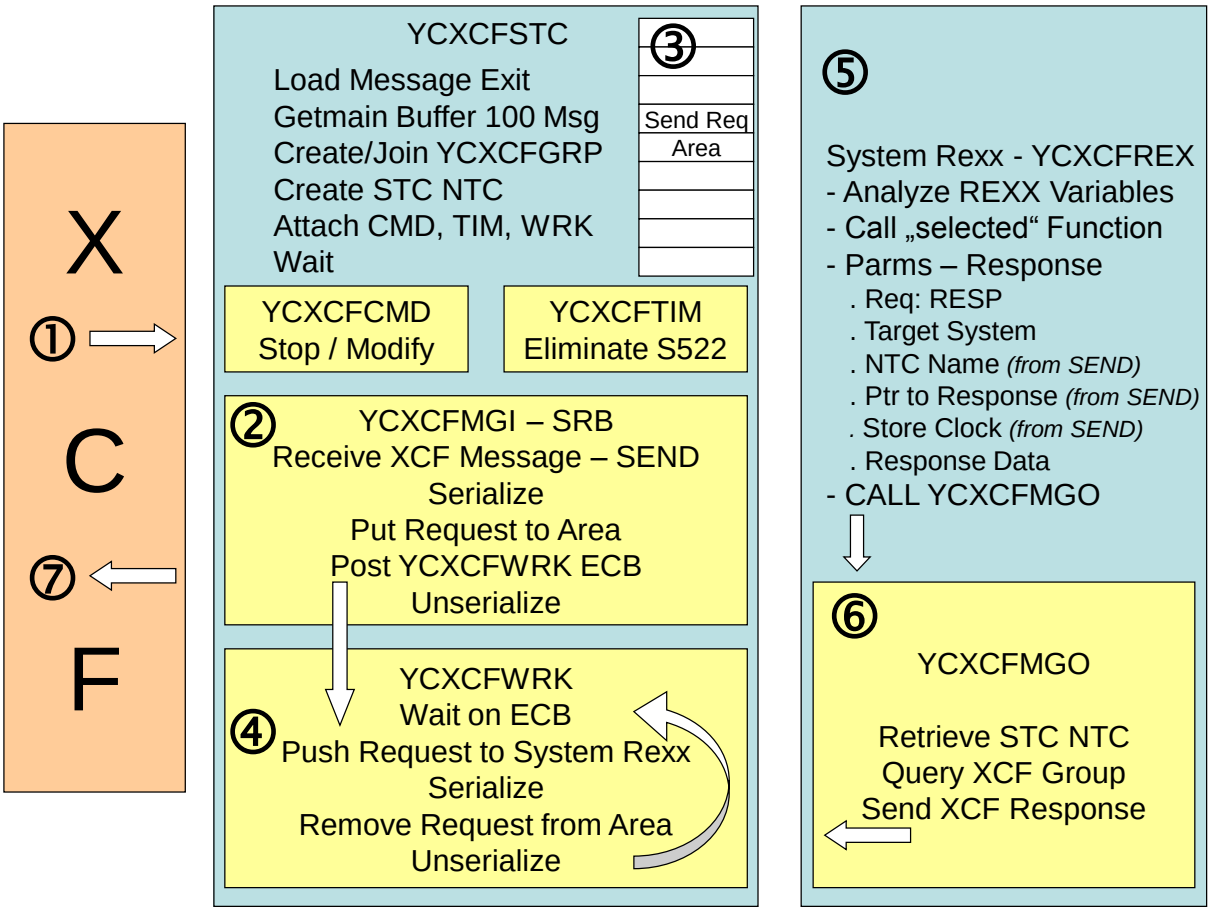


Figure 3: receiving the request

1. XCF will transport the request from the sender YCPlex member to the target YCPlex member; XCF will post the YCPlex message-exit YCXCFMGI
2. The YCXCFMGI message exit:
    a. Read the message
    b. serialize *(Local Lock)* the work
    c. put the message in a buffer area
    d. the request work ECB will be posted
    e. the serialization will be freed
3. the getmain area was aquired by YCXCFSTC
4. YCXCFWRK receives control:
    a. The request will be analyzed and a system rexx request will be created – The Rexx YCXCFREX will be called with following parameter
        i. Asynchron
        ii. Without time limit

Copyright YCOS Yves Colliard Software GmbH

iii. with TSO environment
iv. with the RACF environment of the caller

b. serialization *(Local Lock)*
c. the request will be deleted from the work area
d. serialization will be freed
e. YCXCFWRK goes back into a wait state

5. The system-REXX YCXCFREX gets control
   a. The 2 variablen – YCXCFIN1 and YCXCFIN2 *(a system rexx – AXREXX – constraint)* – contains the message.
   b. the wanted function is called and the result can be provided using the stack. The result can be up to 896 byte long *(including 2 bytes length)*
   c. System-Rexx YCXCFREX call the YCXCFMGO program to send the response back tot he requestor. Following parameters are needed:
      i. Request: RESP
      ii. Target system – only the request sending system can be specified *(as provided from the SEND request – DO NOT CHANGE IT!)*
      iii. the NTC of the SEND request *(as provided from the SEND request – DO NOT CHANGE IT!)*
      iv. the pointer to the to the response area in the requestor address space
      v. der Pointer zur Response im Ziel-Address-Space *(as provided from the SEND request – DO NOT CHANGE IT!)*
      vi. the response-request-area – it must contain a length information *(2 Bytes – including length field!)*. The request can *(including length field)* be up to 896 bytes long
6. the sender programm YCXCFMGO receives control and executes following actions:
   a. the NTC *(Name Token Callable Services)* of the started task will be used
   b. the XCF group will be queried
   c. the response will sent over XCF
7. XCF send the response over CTC *(Channel To Channel)* or CF *(Coupling Facility)* to the requesting YCPlex

1.2.3 Receiving the response – RESP

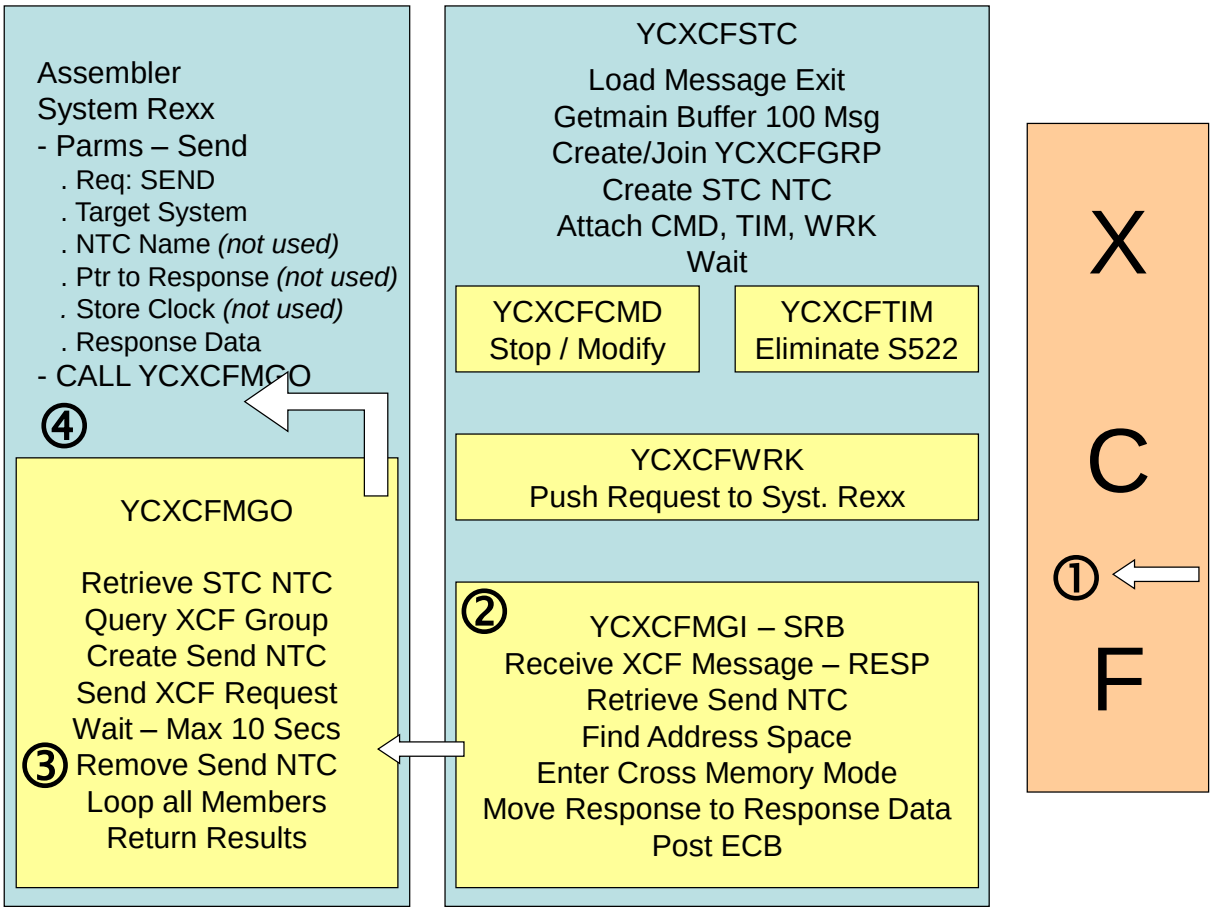
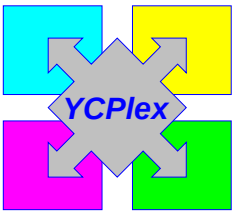


Figure 4: receiving the response

1. XCF will transport the request from the sender YCPlex member to the target YCPlex member; XCF will post the YCPlex message-exit YCXCFMGI
2. The YCXCFMGI message exit:
   a. Read the message
   b. Get the NTC of the request
   c. Find the target address space
   d. Uses cross-memory to copy the response in the answer area
   e. the YCXCFMGO-ECB will be posted
3. YCXCFMGO gets control back
   a. The requestor NTC will be deleted
   b. The next member can be worked
      – at the end of all members
   c. The control goes back the the caller
4. The caller receives the response and can analyze it:

Copyright YCOS Yves Colliard Software GmbH

a. Return code > 4: severe error
b. Return code = 4: system not found
c. Return code = 0 – the response area is filled:
    i. 4 Bytes – number of answers
    ii. For every answer:
        1. 8 Bytes – system name
        2. 2 Bytes – response length *(inkl. the 2 Bytes length field!)* – a 0000 length can be interpreted as a timeout
        3. nnn Bytes – answer – maximum 894



1.3 YCPlex samples

This chapter presents three examples of calling YCXCFMGO; all samples are located in the SAMPLE file or included in the delivered REXX.
The caller must place in the first 8 bytes of the parameter of a request code; this request code is the name of the system rexx which has to be invoked by YCXCFREX on the target system.

1.3.1 System Rexx sample – YCXCFSND

To use this sample, YCXCFSND, YCXCFMRG and YCXCFTST must be installed in system rexx – copy them to SYS1.SAXREXEC or a concatenation.

Calling the sample – using system rexx:

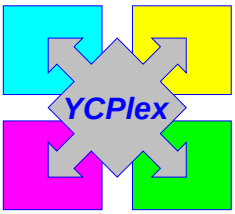
- F AXR,YCXCFSND
  - Without parameter: the text parameter „YCXCFRMG MY SEND information“ will be used
  - The response „MY RESPONSE! hh:mm.ss.hhhhhhh“ will be delivered
- F AXR,YCXCFSND YCXCFTST
  - with „YCXCFTST“ as a parameter: the text parameter „YCXCFTST 123456789!123…“ – length 894 bytes *(maximum)* will be used
  - the response „abcdefghi!abcdefghi!a…“ will be delivered
- F AXR,YCXCFSND xxxxxxxx
  - with „xxxxxxxx“ as a parameter: the text parameter „xxxxxxxxx“ will be used – this is the name of the called system rexx! This one has to be created!
  - currently „Request unknown!“ would be delivered

Copyright YCOS Yves Colliard Software GmbH

Sample – YCXCFSND:

```
/* REXX YCXCFSND Copyright YCOS Yves Colliard Software 2009-10 V2R0  */
/*--------------------------------------------------------------------*/
/* System REXX - Send a request to one or more systems                */
/*               Sample                                               */
/*--------------------------------------------------------------------*/
/* This System Rexx will be called with following commands:           */
/* F AXR,YCXCFSND                                                     */
/*   - Call YCXCFRMG Function                                         */
/* F AXR,YCXCFSND YCXCFTST                                            */
/*   - Call YCXCFTST Function                                         */
/* F AXR,YCXCFSND .other.                                             */
/*   - Call any other Function                                        */
/*--------------------------------------------------------------------*/
/*  V2.0 - 20.07.2010 - new version                                   */
/*  V1.2 - 29.09.2009 - delivery                                      */
/*  P002 - 20.05.2009 small changes                                   */
/*  P001 - 14.05.2009 change name of REXX YCXCFMGO                    */
/*                    new call YCXCFMGO                               */
/*                    new YCXCFMGO Parameter                          */
/*  V1.1 - 07.05.2009 Change parameter Output                         */
/*  V1.0 - 30.04.2009                                                 */
/********************************************************************/
/* Number of Systems - for response buffer */
anzsys=12
/* */
/* Data to be sent */
select
when arg(1)="YCXCFTST" then do
  /* Test with 896!!  */
  prm=left("YCXCFTST"copies("123456789!",90),894)
end
when arg(1)="" then do
  prm='YCXCFRMG MY SEND information'
end
otherwise
  prm=translate(arg(1))
end
prml= length(prm)+2
prm = x2c(d2x(prml,4))''prm
/* Target System */
syst = MVSVAR("SYSNAME")    /* send to this system       */
syst = "*OTHER"             /* send to all other systems */
syst = "*ALL"               /* send to all systems       */
/* */

PARM='...ASEND'         /* Only for YCOS Tests          */
PARM='SEND'               /* SEND/RESP                  */
PARM=PARM''left(syst,8)   /* Target System              */
PARM=PARM''left(' ',16)   /* orig NTC TToken            */
PARM=PARM''left(' ',4)    /* orig Ptr to Resp           */
PARM=PARM''left(' ',8)    /* orig Store Clock           */
wo=length(parm)+4+1+prml  /* Lgt + Position + Parameter*/
```




```
PARM=PARM''Left(prm,(anzsys+1)*(896+8)) /*the responses*/
t_lgt = x2c(d2x(length(PARM)+4,8))
SYSAUTH_YCXCFINP=t_lgt""PARM /* length + PARM       */
/* Call the Assembler Program */
yrc=outtrap("ycxcf.")
address tso "CALL *(YCXCFMGO)"
yrc=outtrap("OFF")
/* */
do i=1 to ycxcf.0
  yrc = AXRWTO(ycxcf.i)
end
/* */
if rc=0 then do
  /* only needed for debugging  */
  /* issue control message      */
  p_num = substr(SYSAUTH_YCXCFOUT,wo,4) /* number of Responses */
  p_num = x2d(c2x(p_num))
  lin="YXCFD01I Responses received:" p_num
  ml_id="FIRSTLINE"
  yrc = AXRMLWTO(lin,'ml_id',"L")
  if p_num= 0 then do
    yrc = AXRMLWTO(,'ml_id',"E")
  end
  else do
    wo = wo+4
    do i=1 to p_num
      p_sys = substr(SYSAUTH_YCXCFOUT,wo,8)
      p_lgt = substr(SYSAUTH_YCXCFOUT,wo+8,2)
      p_lgt = x2d(c2x(p_lgt))
      lin = "System:" substr(SYSAUTH_YCXCFOUT,wo,8)
      yrc = AXRMLWTO(lin,'ml_id',"D")
      if i=p_num then lc="DE"
      else lc="D"
      if p_lgt=0 then do
        lin = "Error Time out - no response within 10 seconds"
        p_lgt=2
      end
      else do
        p_prm = substr(SYSAUTH_YCXCFOUT,wo+10,p_lgt-2)
        /* if left(p_prm,5)="YXCFR" then -> Message from YCXCFREX */
        if p_lgt>42 then do
          inf=left(p_prm,40)"..."
          inf=inf""right(p_prm,min(3,p_lgt-42))
        end
        else do
          inf=left(p_prm,p_lgt-2)
        end
        lin = right(p_lgt,3) "Bytes RESP:" inf
      end
      yrc = AXRMLWTO(lin,'ml_id',lc)
      wo = wo + 10 + p_lgt - 2
    end
  end
end
```


Copyright YCOS Yves Colliard Software GmbH

```
else do
  yrc = AXRWTO("YXCFD02A YCXCFMGO Return Code" rc)
end
/* */
exit
```

Figure 5: sample system rexx – request – YCXCFSND

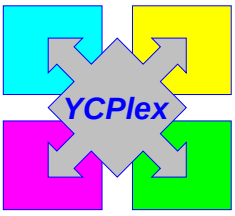
Sample – YCXCFRMG:

```
/* REXX YCXCFMGO Copyright YCOS Yves Colliard Software 2009-10       */
/*-------------------------------------------------------------------*/
/* System REXX - Sample System Rexx Request Handler                  */
/*             - get control from YCPLEX - YCXCFREX                  */
/*-------------------------------------------------------------------*/
/*  V2.0 - 20.07.2010 - new version                                  */
/*  P005 - 12.10.2009 - Support for Userid propagation               */
/*  V1.2 - 29.09.2009 - delivery                                     */
/*  P003 - 15.07.2009 - change receiving parameter                   */
/*  P002 - 20.05.2009 - change return response                       */
/*  P001 - 14.05.2009 - first delivery                               */
/*-------------------------------------------------------------------*/
/* Create a response                                                 */
/*-------------------------------------------------------------------*/
parse pull ycxcfinf
MSG_DIAG =substr(arg(1),1,32)           /* extract Diagnosis         */
MSG_LEN  =substr(ycxcfinf, 1,4)         /*         length            */
MSG_INFO =substr(ycxcfinf, 5,4)         /*         info: SEND        */
MSG_STCK =substr(ycxcfinf, 9,8)         /*         Store Clock       */
MSG_SYSN =substr(ycxcfinf,17,8)         /*         sending System    */
MSG_USER =substr(ycxcfinf,25,8)         /*         sending User Id   */
MSG_NTOK =substr(ycxcfinf,33,16)        /*         sending Name Tok  */
MSG_PTRR =substr(ycxcfinf,49,4)         /*         sending Data Prt  */
MSG_PARM =substr(ycxcfinf,53)           /* Length and Data           */
MSG_PLGT =x2d(c2x(left(MSG_PARM,2)))    /*   length of Data          */
MSG_PREQ =substr(ycxcfinf,55,8)         /*   Requested Function      */
MSG_PDAT =substr(ycxcfinf,63)           /*   Parameter Data          */
/* work the data */
queue 'MY RESPONSE!' time('L')     /* the response     */
/* */
exit 0
```

Figure 6: sample system rexx – response – YCXCFMRG

Sample – YCXCFTST:

```
/* REXX YCXCFTST Copyright YCOS Yves Colliard Software 2009-10       */
/*-------------------------------------------------------------------*/
/* System REXX - Sample System Rexx Request Handler                  */
/*             - get control from YCPLEX - YCXCFREX                  */
/*-------------------------------------------------------------------*/
/*  V2.0 - 20.07.2010 - new version                                  */
/*  P005 - 12.10.2009 - Support for Userid propagation               */
/*  V1.2 - 29.09.2009 - delivery                                     */
```




```
/*  P003 - 15.07.2009 - change receiving parameter                    */
/*  P002 - 20.05.2009 - change return response                        */
/*  P001 - 14.05.2009 - first delivery                                */
/*--------------------------------------------------------------------*/
/* Create a long response                                             */
/*--------------------------------------------------------------------*/
parse pull ycxcfinf
MSG_DIAG =substr(arg(1),1,32)           /* extract Diagnosis          */
MSG_LEN  =substr(ycxcfinf, 1,4)         /*         length             */
MSG_INFO =substr(ycxcfinf, 5,4)         /*         info: SEND         */
MSG_STCK =substr(ycxcfinf, 9,8)         /*         Store Clock        */
MSG_SYSN =substr(ycxcfinf,17,8)         /*         sending System     */
MSG_USER =substr(ycxcfinf,25,8)         /*         sending User Id    */
MSG_NTOK =substr(ycxcfinf,33,16)        /*         sending Name Tok   */
MSG_PTRR =substr(ycxcfinf,49,4)         /*         sending Data Prt   */
MSG_PARM =substr(ycxcfinf,53)           /* Length and Data            */
MSG_PLGT =x2d(c2x(left(MSG_PARM,2)))    /*   length of Data           */
MSG_PREQ =substr(ycxcfinf,55,8)         /*   Requested Function       */
MSG_PDAT =substr(ycxcfinf,63)           /*   Parameter Data           */
/* work the data */
queue left(copies("abcdefghi!",90),894)   /* Test with 896!!  */
exit 0
```

Figure 7: sample system rexx – response – YCXCFTST

1.3.2 Sample – YCPLEXQY

The rexx sample YCPLEXQY can be used to check the sysplex YCPlex query – QURY – function.
The rexx can be called – if authorized – unter TSO and also in batch:

- YCPLEXQY

Calling YCPLEXQY with the BATCH parameter will display the JCL needed to run the sample in batch.

Sample output:

```
   Menu  Utilities  Compilers  Help
 ------------------------------------------------------------------------------
 BROWSE    SYS09217.T124911.RA000.YVES.R0100548       Line 00000000 Col 001 080
 Command ===>                                           Scroll ===> CSR
******************************** Top of Data ********************************
YCplex Group Query - YCXCFGRP - in Sysplex PERPLEX

Responses received: 1

System: SYSY     running YCPlex


(C) YCOS Yves Colliard Software GmbH 2009-09
******************************** Bottom of Data ********************************
```

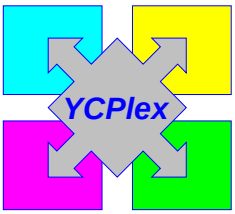

Copyright YCOS Yves Colliard Software GmbH

1.3.3 Assembler sample – YCXCFASM

The sample assembler program YCXCFASM invokes YCXCFMGO with the YCXCFRMG parameter. All answers are issued per message.

The sample YCXCFASM must be installed in an APF authorized library *(compile and bind)*. YCXCFASM needs the MGOPARM macro.

The program call can be done using a batch job – the sample YCXCFASJ can be used.



2 Installation

See YCSLOG – YCOS System Logger Utility User's Guide Manual.

2.1 STC: Started Task

The started task must be copied into the PROCLIB concatenation *(YCPlex name can be used for example)*. Using a STEPLIB is optional and must only be done if the used file was not included the LINKLST concatenation *(PROG members)*.

```
//*-------------------------------------------------------
//* YCPLEX - YCOS PLEX COMMUNICATION
//* COPYRIGHT YCOS YVES COLLIARD SOFTWARE GMBH - 2009-10
//*-------------------------------------------------------
//YCPLEX   EXEC PGM=YCXCFSTC  ,PARM='HLQDIAG'
//*                            HLQ OF SYSTEM REXX DIAG DATASET DIAGREXX
//STEPLIB  DD DISP=SHR,DSN=HLQ.YCSLOG.LINKAPF
//SYSUDUMP DD SYSOUT=*
```

Figure 8: YCPlex started task

Change the PARM if an other HLQ for diagnosis datasets are needed – see YCXCFREX System Rexx Diagnosis page 25

2.2 RACF

Using the YCXCFMGO interface is subject to protection to RACF. Following profiles are used:

- SEND requests
  - Class FACILITY
  - Profile: YCPLEX.SEND.function.sysname
- RESP Requests
  - Class FACILITY
  - Profile: YCPLEX.RESP

The SEND request can be allowed per function/system with a READ permission.

The RESP request must authorize the SEND users with a READ permission.

The QURY request is not protected.

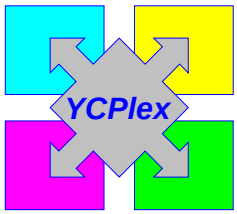
Copyright YCOS Yves Colliard Software GmbH

2.3 Performance

The started task should be at least be classified to a medium service class of WLM.

Attention: the AXR and the AXR## address spaces needs also a medium classification.

Copyright YCOS Yves Colliard Software GmbH



3 Operation

3.1 Start of YCPlex

Start YCPlex using following MVS start command:

```
S YCPLEX
```

3.2 Stop of YCPlex

YCPlex can be stopped using following MVS stop- oder modify commands:

```
P YCPLEX
or
F YCPLEX,SHUTDOWN
or
F YCPLEX,STOP
or
F YCPLEX,P
```

3.3 Modules, Versionen, PTF and Compile of YCPlex

A command is also available, to display the modules YCPlex. Also versions of PTF status and compile information displayed. This information can be very important during debugging; here an example:

```
F YCPLEX,LOADMODS

YXCFC04I YCXCFSTC,COPYRIGHT YCOS GMBH 2009-10,V2R0,PTF0010,07/05/10,21.34
YXCFC05I YCXCFMGI,COPYRIGHT YCOS GMBH 2009-10,V2R0,PTF0010,07/05/10,20.59
YXCFC04I YCXCFTIM,COPYRIGHT YCOS GMBH 2009-10,V2R0,PTF0010,07/05/10,22.01
YXCFC04I YCXCFCMD,COPYRIGHT YCOS GMBH 2009-10,V2R0,PTF0010,07/05/10,08.26
YXCFC05I YCXCFMGO,COPYRIGHT YCOS GMBH 2009-10,V2R0,PTF0010,07/05/10,21.07
YXCFC04I YCXCFWRK,COPYRIGHT YCOS GMBH 2009-10,V2R0,PTF0010,07/05/10,22.06
```


Copyright YCOS Yves Colliard Software GmbH

3.4 YCXCFREX System Rexx Diagnosis

An output dataset can be created from the YCXCFREX system REXX for diagnostic purposes; it will contain SAY, TRACE and errors. The current state of diagnosis is obtained using the following command:

```
F YCPLEX,DIAGREXX
YXCFC08I DIAGREXX CURRENTLY OFF
or
YXCFC07I DIAGREXX CURRENTLY ON
```

The diagnostic dataset can be turned on or off by using following command:

```
F YCPLEX,DIAGREXX=ON
YXCFC09I DIAGREXX SET TO ON
or
F YCPLEX,DIAGREXX=OFF
YXCFC10I DIAGREXX SET TO OFF
```

Any YCXCFREX call will create an output file with following name:
hlq.YCPLEX.sysname_exec.sysname_send.Dyymmdd.Thhmmss

- the hlq as given on the started task *(PARM)* – Default SYS1
- sysname_exec – the executing system name
- sysname_send – the requestor system name

The YCPlex user needs RACF Alter access to the datasets!

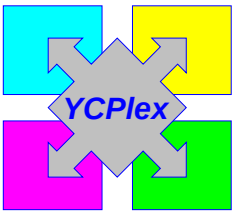
A multi line message can also be activated for the YCXCFREX system REXX for diagnostis purposes; it will produce message YXCFR01I. The current state of diagnosis is obtained using the following command:

```
F YCPLEX,DIAGRMSG
YXCFC12I DIAGRMSG CURRENTLY OFF
or
YXCFC11I DIAGRMSG CURRENTLY ON
```

The diagnostic message can be turned on or off by using following command:

```
F YCPLEX,DIAGRMSG=ON
YXCFC13I DIAGRMSG SET TO ON
or
F YCPLEX,DIAGRMSG=OFF
YXCFC14I DIAGRMSG SET TO OFF
```

Diagnosis settings are reset to OFF at the stop of the started task.



3.5 Messages of YCPlex

The messages of YCPlex have folling format:

```
YXCFx##y
!   !! !
!   !! +-- I=Info
!   !!     E=Error
!   !!     A=Error
!   !!
!   !+-- Numbering
!   !
!   +-- Module/Task/Rexx containing/producing the message:
!       C=YCXCFCMD - Command Interface
!       T=YCXCFTIM - Abend 522 exclusion
!       S=YCXCFSTC - Main Started Task
!       W=YCXCFWRK - Work Task - AXREXX Interface
!       O=YCXCFMGO - Message Out
!       I=YCXCFMGI - Message In
!       R=YCXCFREX - System Rexx Message In Interface
!       D=YCXCFSND - System Rexx Sample
!       A=YCXCFASM - Assembler Sample
+-- Message Id - YXCF
```


Copyright YCOS Yves Colliard Software GmbH

3.5.1 Messages of YCXCFCMD:

YXCFC01A ECB MUST BE ZERO

Internal error. YCPlex will stop all activities – User Abend 1112.
Review the messages and dump, call YCOS support

YXCFC02E UNKNOWN MODIFY PARAMETER=

entered modify parameter is incorrect. supports only SHUTDOWN, STOP or P or LOADMODS

YXCFC03I STOPPING YCPLEX

a stop or modify stop command is entered. YCPlex will stop all activity.

YXCFC04I YCXCFxxx,COPYRIGHT YCOS GMBH jjjj-jj,VxRy,PTFnnnn ,tt/mm/jj,hh.mm

Modify LOADMODS command output – display of the YCPlex started task modules attached

YXCFC05I YCXCFxxx,COPYRIGHT YCOS GMBH jjjj-jj,VxRy,PTFnnnn ,tt/mm/jj,hh.mm

Modify LOADMODS command output – display of the YCPlex started task modules loaded

YXCFC06I P,STOP,SHUTDOWN,LOADMODS

(F) modify command unknown – list of available commands – DIAGREXX will not be listed

YXCFC07I DIAGREXX CURRENTLY ON

(F) modify the DIAGREXX command – this option is ON at this time

YXCFC08I DIAGREXX CURRENTLY OFF

(F) modify the DIAGREXX command – this option is OFF at this time

YXCFC09I DIAGREXX SET TO ON

Command DIAGREXX = modify (F) ON executed – the option is now active

YXCFC10I DIAGREXX SET TO OFF

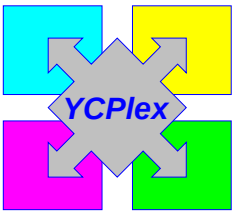
Command DIAGREXX = modify (F) OFF executed – the option is now inactive

YXCFC11I DIAGRMSG CURRENTLY ON

(F) modify the DIAGRMSG command – this option is ON at this time

YXCFC12I DIAGRMSG CURRENTLY OFF

(F) modify the DIAGRMSG command – this option is OFF at this time



YXCFC13I DIAGRMSG SET TO ON
Command DIAGRMSG = modify (F) ON executed – the option is now active

YXCFC14I DIAGRMSG SET TO OFF
Command DIAGRMSG = modify (F) OFF executed – the option is now inactive

3.5.2 Messages of YCXCFTIM:

YXCFT01A ECB MUST BE ZERO
Internal error. YCPlex will stop all activities – User Abend 1113.
Review the messages and dump, call YCOS support

Copyright YCOS Yves Colliard Software GmbH

3.5.3 Messages of YCXCFSTC:

YXCFS01I YCPLEX COPYRIGHT YCOS GMBH jjjj-jj, VxRy
YCPlex is starting

YXCFS02I YCPLEX STARTING WITH GROUP xxxxxxxx
XCF group name

YXCFS03I XCF GROUP xxxxxxxx yyyyyyy BY zzzzzzzz ON ttttttt
YCPlex has created/joined member zzzzzzzz of XCF group xxxxxxxx on system ttttttt. Member zzzzzzzz is the first member in the group – yyyyyyy CREATED – or has joined the group – yyyyyyy JOINED

YXCFS04I YCPLEX INITIALIZATION COMPLETE
YCPlex is up and running

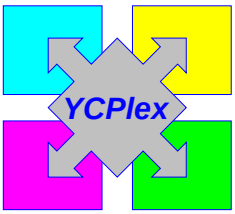
YXCFS05I YCPLEX STOPPED
YCPlex was stopped

YXCFS06A YCPLEX STOPPED - WITH ERRORS
Internal error. YCPlex has stopped without a stop command all activities *(Subtask Error!)* – User Abend 1111.
Review the messages and dump, call YCOS support

YXCFS07A PARAMETER LENGTH ERROR
Start PARM – parameter – error

YXCFS08A GROUP/MEMBER ALREADY ACTIVE ON THIS SYSTEM
Error: XCF group/member is already active on this system

YXCFS09A ERROR DURING XCF xxxxx,RETURN CODE=yyy,REASON CODE=zzzz
General error calling an MVS interface:
LOADI – error loading YCXCFMGI
JOIN1 – error during IXCJOIN
ATTIM – error during ATTACH of YCXCFTIM
ATCMD – error during ATTACH of YCXCFCMD
ATWRK – error during ATTACH of YCXCFWRK
NTCCR – error during Name Token Create
Internal error. YCPlex will stop all activities – User Abend 1114.
Review the messages and dump, call YCOS support



3.5.4 Messages of YCXCFWRK:

YXCFW01A ECB MUST BE ZERO

Internal error. YCPlex will stop all activities – User Abend 1115.
Review the messages and dump, call YCOS support

YXCFW02A POINTER TO WORK AREA ZERO

Internal error. YCPlex will stop all activities – User Abend 1116.
Review the messages and dump, call YCOS support

YXCFW03A ERROR DURING XCF xxxxx,RETURN CODE=yyy,REASON CODE=zzzz

General error calling an MVS interface:
AXRX1 – error during AXREXX
Internal error. YCPlex will stop all activities – User Abend 1117.
Review the messages and dump, call YCOS support

Copyright YCOS Yves Colliard Software GmbH

3.5.5 Messages of YCXCFMGO:

YXCFO01A PARAMETER LENGTH ERROR
Parameter Error

YXCFO02A PARAMETER REQUEST ERROR: SEND/RESP
The request must be SEND or RESP

YXCFO03A PARAMETER - SEND TO - ERROR
The target name is wrong:
during SEND: „*OTHER“, „*ALL“ or System-Name expected
during RESP: System-Name expected

YXCFO04A GROUP/MEMBER NOT ALREADY ACTIVE ON THIS SYSTEM
YCPlex not already started – start YCPlex and retry

YXCFO05A PARAMETER ERROR - LENGTH OF DATA GREATER nnn
Current maximum request length greater than nnn *(896)*.

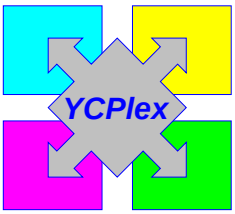
YXCFO06A TIME OUT - NO RESPONSE FROM SYSTEM xxxxxxxx
Time out reached – default 10 seconds or the defined time out

YXCFO07E NO RESPONSE RECEIVED
The selected systems *(„*OTHER“ or System-Name)* were not found or no response were delivered

YXCFO08E NOT ENOUGH SPACE IN BUFFER TO HOLD ALL RESPONSES
Response parameter area too small

YXCFO09A ERROR DURING XCF xxxxx,RETURN CODE=yyy,REASON CODE=zzzz
General error calling an MVS interface:
QURY1 – error during IXCQUERY
MSGO1 – error during IXCMSGO
NTCCR – error during Name Token Create
NTCDL – error during Name Token Delete
Internal error. YCPlex will stop all activities – User Abend 1118.
Review the messages and dump, call YCOS support

YXCFO10A ACCESS DENIED TO PROFILE xxxxxxxx....
The SEND and RESP functions of YCXCFMGO are RACF protected – authorization not granted – review the profiles and access authorities



3.5.6 Messages of YCXCFMGI:

YXCFI02A POST ERROR CODE 8 FROM SYSTEM xxxxxxxx
Post Error Code 8 – Call YCPlex Support

YXCFI03A POST ERROR ERRET FROM SYSTEM xxxxxxxx
Post Erret Error – Call YCPlex Support

YXCFI04A NAME TOKEN ERROR - NO RESPONSE FROM SYSTEM xxxxxxxx
Name Token for response not found – it could be a parameter error or the requestor address space endet before the response has arrived

YXCFI05A ASCB CONTROL BLOCK EYECATCHER ERROR FROM SYSTEM xxxxxxxx
ASCB control block not found – it could be a parameter error or the requestor address space endet before the response has arrived

YXCFI06A STCK COMPARE ERROR FROM SYSTEM xxxxxxxx
Store Clock of SEND and RESP are not correct – it could be a parameter error or the requestor address space endet before the response has arrived

YXCFI07A ERROR DURING XCF xxxxx,RETURN CODE=yyy,REASON CODE=zzzz
General error calling an MVS interface:
STOR1 – error during STORAGE OBTAIN
MSGI1 – error during IXCMSGI
NTCRT – error during Retrieve of Name Token
Call YCPlex Support

Copyright YCOS Yves Colliard Software GmbH

3.5.7 Messages of REXX – YCXCFREX:

YXCFR01I Rec: hh:mm:ss.hhhhhh Snd: dd/mm/yyyy hh:mm:ss.hhhhhh
xxxx System: xxxxxxxx User Id : xxxxxxxx
Resp. Ptr : xxxxxxxx NTC Token: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
nnn Bytes PARM: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Rec: hh:mm:ss.hhhhhh – Message receive time

Snd: dd/mm/yyyy hh:mm:ss.hhhhhh – Message send date and time

xxxx System: xxxxxxxx User Id : xxxxxxxx – xxxx Action *(SEND)* – Sending System and User Id

Resp. Ptr : xxxxxxxx NTC Token: x – Response pointer and NTC Token in sending address space

nnn Bytes SEND: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx – length and up to 40 bytes of sended data

Debugging information – DIAGRMSG

YXCFR02A YCXCFMGO Return Code
Error during call of YCXCFMGO

YXCFR03A Problem during STCK conversion
Error during Store Clock conversion

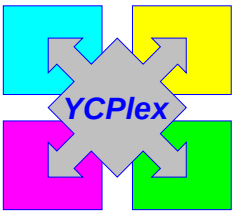
YXCFR04A Request unknown request_code
During a SEND request a wrong parameter – system rexx name – was given

YXCFR05A Response length exceeded number_bytes
Response exceeds the supported size

YXCFR06A Return Code xx from function yyyyyyyy
Error during call yyyyyyyy system rexx. Review if the system rexx yyyyyyyy is in SYS1.SAXREXEC or concatenation

YXCFR07A YXCFR07A no data returned from function yyyyyyyy
Error during call yyyyyyyy system rexx. Review system rexx yyyyyyyy – check if a „QUEUE“ of the response is done! Stack is empty!

Copyright YCOS Yves Colliard Software GmbH



3.5.8 Messages of REXX – YCXCFSND:

YXCFD01I Responses received: nn
System: xxxxxxxx
nnn Bytes RESP: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Responses received – Number of responses received

System: xxxxxxxx – Response from System

nnn Bytes RESP: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx – length and up to 40 bytes of responded data

Debugging information

YXCFD02A YCXCFMGO Return Code

Error during call of YCXCFMGO

Copyright YCOS Yves Colliard Software GmbH

3.5.9 Messages von Assembler – YCXCFASM:

YXCFA01I SYSTEM: xxxxxxxx RESPONSE:
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

System Name
up to 40 bytes of responded data

YXCFA02I SYSTEM: xxxxxxxx NO RESPONSE

No response from system xxxxxxxx

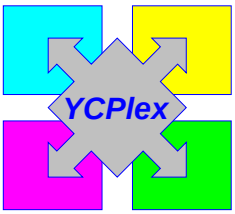
YXCFA03I YCXCFMGO RETURN CODE 4 - NO SYSTEM FOUND

YCXCFMGO returned a return code 4

YXCFA04A YCXCFMGO RETURN CODE 8 - CATASTROPHIC ERROR

YCXCFMGO returned a return code 8 – User Abend 1119

Copyright YCOS Yves Colliard Software GmbH



4 YCPlex Support

YCOS Yves Colliard Software GmbH
Fremersbergstr. 45
D-76530 Baden-Baden

Tel: (D) 07221/9708384
Fax: (D) 0322 2374 2352

e-Mail: ycos@ycos.de
Home: http://www.ycos.de

Copyright YCOS Yves Colliard Software GmbH

5 YCPlex Version and Release

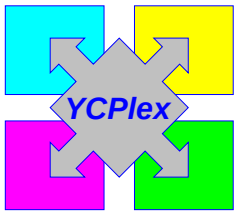
5.1 Version 2 Release 0 and PTFs

PTF Mod	Date - Comment
V2.0	Planed 31.10.10 – Version 2.0 - GA – General Availability

5.2 Version 1 Release 2 and PTFs

PTF Mod	Date - Comment
V1.2	02.10.09 – Version 1.2 – YCPLex in YCSLOG integriert
P004 YCXCFMGO Loadlib	08.10.09 – Version 1.2 – PTF0004 - Support for other Wait Interval
P005 YCXCFMGO Loadlib	12.10.09 – Version 1.2 – PTF0005 - Support for Userid propagation
P005 YCXCFMGO Loadlib	12.10.09 – Version 1.2 – PTF0005 - Support for Userid propagation

Copyright YCOS Yves Colliard Software GmbH

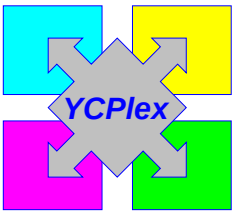


5.3 Version 1 Release 1 and PTFs

PTF Mod	Date - Comment
P003 MGOPARM Sample	03.08.09 – Version 1.1 – PTF0003 - new request code QURY
P003 YCXCFASM Sample	03.08.09 – Version 1.1 – PTF0003 - correct work of parms
P003 YCXCFMGO Loadlib	03.08.09 – Version 1.1 – PTF0003 - If no TSO Var check for Parameter - change PUTLINE check - new QURY Parameter
P002 YCXCFTST Sample	20.05.09 – Version 1.1 – PTF0002 - change return response
P002 YCXCFSND Sample	20.05.09 – Version 1.1 – PTF0002 - small changes
P002 YCXCFMRG Sample	20.05.09 – Version 1.1 – PTF0002 - change return response
P002 YCXCFREX Rexx	20.05.09 – Version 1.1 – PTF0002 - correct MSG_PREQ
P002 YCXCFSTJ Sample	20.05.09 – Version 1.1 – PTF0002 - support a HLQ qualifier for Diagnosis Rexx dataset
P002 YCXCFWRK Loadlib	20.05.09 – Version 1.1 – PTF0002 - Correct SETLOCK OBTAIN save - Change Message Dataset name
P002 YCXCFSTC Loadlib	20.05.09 – Version 1.1 – PTF0002 - support a HLQ qualifier for Diagnosis Rexx dataset
P002 YCXCFMGI Loadlib	20.05.09 – Version 1.1 – PTF0002 - Correct SETLOCK OBTAIN save
P001 YCXCFREX Rexx	14.05.09 – Version 1.1 – PTF0001 - new variable YCXCFDIA - Diagnose - create REXX Trace on DIAGREXX - create Message on DIAGRMSG

Copyright YCOS Yves Colliard Software GmbH

PTF Mod	Date - Comment
	- new call YCXCFMGO - new YCXCFMGO Parameter
P001 YCXCFASM Sample	14.05.09 – Version 1.1 – PTF0001 - change of Parameters
P001 YCXCFSND Sample	14.05.09 – Version 1.1 – PTF0001 - change name of REXX YCXCFMGO - new call YCXCFMGO - new YCXCFMGO Parameter
P001 YCXCFTST Sample	14.05.09 – Version 1.1 – PTF0001 - first delivery
P001 YCXCFRMG Sample	14.05.09 – Version 1.1 – PTF0001 - first delivery
P001 MGOPARM Sample	14.05.09 – Version 1.1 – PTF0001 - change of Parameters
P001 YCXCFCMD Loadlib	14.05.09 – Version 1.1 – PTF0001 - new Modify Command – DIAGRMSG - new Messages YXCFC11I - YXCFC15I - Load YCXCFMGO during LOADMODS
P001 YCXCFMGI Loadlib	14.05.09 – Version 1.1 – PTF0001 - Ignore POST Error Code 4 - Msg YXCFI01A - Change Response Error messages - put Sysname - Change NTC Error message DESC Code - Compare Store Clock on Response - Msg YXCFI06A
P001 YCXCFMGO Loadlib	14.05.09 – Version 1.1 – PTF0001 - Check if TSO - get Parms from variable - set Response to variable SYSAUTH_YCXCFINP and YCXCFOUT - putline the errors (not WTO) - Check RACF Facility Class YCPLEX.SEND/RESP Profs - Change in Parameters - Check of Store Clock - Put System name in message YXCFO06A
P001 YCXCFSTC Loadlib	14.05.09 – Version 1.1 – PTF0001 - small changes
P001 YCXCFWRK Loadlib	14.05.09 – Version 1.1 – PTF0001 - Create New Variable for Diagnosis Messages - Security by AXR User



PTF Mod	Date - Comment
V1.1 MGOPARM	07.05.09 – Version 1.1 - change of Parameters
V1.1 YCXCFASM	07.05.09 – Version 1.1 - 0C4 wegen falsche Berechnung - neue No Response Meldung - Change der Output Parameter
V1.1 YCXCFCMD	07.05.09 – Version 1.1 - Change in Parameter - new Modify Command - DIAGREXX
V1.1 YCXCFMGI	07.05.09 – Version 1.1 - Change in Parameter Name - Change in Work Area
V1.1 YCXCFMGO	07.05.09 – Version 1.1 - Change in Parameter Name - Change in Output Parameter - Change der WTO wegen AXR address space
V1.1 YCXCFREX	07.05.09 – Version 1.1 - change in parameter
V1.1 YCXCFSND	07.05.09 – Version 1.1 - Change parameter Output
V1.1 YCXCFSTC	07.05.09 – Version 1.1 - YCXCFWRK + YCXCFCMD -> Diag REXX Output Dataset
V1.1 YCXCFWRK	07.05.09 – Version 1.1 - Create Message Dataset bei System Rexx Fehler - Change of Work Area - Diagnosis

5.4 Version 1 Release 0 und PTFs

PTF Mod	Date - Comment
00000	30.04.09 – Version 1.0 - first delivery

Trademarks:

IBM ™ MVS/ESA ™ MVS/XA ™ OS/390 ™
RACF ™ z/OS ™
are trademarks of International Business Machines Corporation.

Copyright YCOS Yves Colliard Software GmbH

6 Index

B___
Beispiel
YCXCFRMG 9

C___
CF Coupling Facility
Channel To Channel see CTC
Compile
YCPlex 24
Coupling Facility see CF
Cross System Coupling Facility see XCF
CTC Channel To Channel

D___
Diagnosis
System Rexx 25
YCXCFREX 25
DIAGRMSG
YCXCFREX 33
YXCFR01I 25

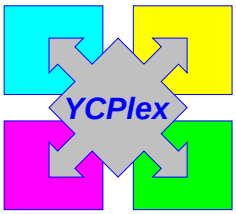
E___
ECB Event Control Block
Event Control Block ECB

F___
F
YCPlex LOADMODS 24
F DIAGREXX
YCPlex 25
F DIAGRMSG
YCPlex 25
FACILITY
RACF 7

I___
Installation
description 22

M___
Message
YXCFA01I 35
YXCFA02I 35
YXCFA03I 35
YXCFA04A 35
YXCFC01A 27
YXCFC02E 27
YXCFC03I 27
YXCFC04I 27
YXCFC05I 27
YXCFC06I 27
YXCFC07I 27
YXCFC08I 27
YXCFC09I 27
YXCFC10I 27
YXCFC11I 27
YXCFC12I 27
YXCFC13I 28
YXCFC14I 28
YXCFD01I 34
YXCFD02A 34
YXCFI02A 32
YXCFI03A 32
YXCFI04A 32
YXCFI05A 32
YXCFI06A 32
YXCFI07A 32
YXCFO01A 31
YXCFO02A 31
YXCFO03A 31
YXCFO04A 31
YXCFO05A 31
YXCFO06A 31
YXCFO07E 31
YXCFO08E 31
YXCFO09A 31
YXCFO10A 31
YXCFR01I 33
YXCFR02A 33
YXCFR03A 33
YXCFR04A 33
YXCFR05A 33
YXCFR06A 33
YXCFR07A 33
YXCFS01I 29
YXCFS02I 29
YXCFS03I 29
YXCFS04I 29
YXCFS05I 29
YXCFS06A 29
YXCFS07A 29
YXCFS08A 29
YXCFS09A 29
YXCFT01A 28
YXCFW01A 30
YXCFW02A 30

Copyright YCOS Yves Colliard Software GmbH



YXCFW03A 30
Messages
YCPlex 26
YCXCFASM 35
YCXCFCMD 27
YCXCFMGI 32
YCXCFMGO 31
YCXCFREX 33
YCXCFSND 34
YCXCFSTC 29
YCXCFTIM 28
YCXCFWRK 30
Modify DIAGREXX
YCPlex 25
Modify DIAGRMSG
YCPlex 25
Modify LOADMODS
YCPlex 24
Module
YCXCFCMD 6
YCXCFMGI 6
YCXCFMGO 7
YCXCFSTC 6
YCXCFTIM 6
YCXCFWRK 6
Modules
YCPlex 24

N___
Name Token Callable Services see NTC
NTC Name Token Callable Services

O___
operation
overview 24
Overview
YCPlex 5

P___
Performance
YCPlexe 23
processing
YCPlex 10
PTF
YCPlex 24

R___
RACF Resouce Access Control Facility
FACILITY 7
Setup 22
YCPlex 7
Receive Request
SEND 12
Receive RESP
YCXCFMGI 14
Receive Response
RESP 14
Receive SEND
YCXCFMGI 12
YCXCFREX 12
YCXCFSTC 12
Release
YCPlex 37
Request Length
RESP 13
SEND 11
RESP
Receive Response 14
Request Length 13
YCXCFMGO 13
REXX
YCXCFREX 9

S___
sample
YCPLEXQY 20
YCXCFASM 21
YCXCFMRG 19
YCXCFSND 16; 17
YCXCFTST 19
samples
YCPlex 16
SEND
Receive Request 12
Request Length 11
Send Request 10
YCXCFCMD 11
YCXCFGRP XCF Group 10
YCXCFMGI 10
YCXCFMGO 11
YCXCFSTC 10
YCXCFTIM 11
YCXCFWRK 11
Send Request
SEND 10
Shutdown
YCPlex 24
Start
YCPlex 24
STC Started Task

Copyright YCOS Yves Colliard Software GmbH

Performance 23
Setup 22
Stop
YCPlex 24
Support
YCPlex 36
System Rexx
Diagnosis 25

T___

Task Control Block TCB
TCB Task Control Block

V___

Version
YCPlex 37
Version V1R0
YCPlex 40
Version V1R1
YCPlex 38
Version V1R2
YCPlex 37
Version V2R0
YCPlex 37
Versionen
YCPlex 24

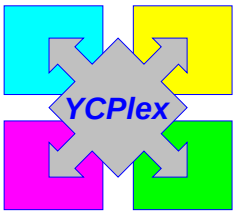
X___

XCF Cross System Coupling Facility
XCF Group
YCXCFGRP 6
XCF Member
YCXCFGRP 6

Y___

YCPlex YCOS Sysplex
Compile 24
F DIAGREXX 25
F DIAGRMSG 25
F LOADMODS 24
Installation 22
Messages 26
Modify DIAGREXX 25
Modify DIAGRMSG 25
Modify LOADMODS 24
Modules 24
operation 24
Overview 5
Performance 23
processing 10
PTF 24
RACF 7; 22
samples 16
Shutdown 24
Start 24
STC 22
Stop 24
Support 36
Version V1R0 40
Version V1R1 38
Version V1R2 37
Version V2R0 37
Versionen 24
YCPLEXQY
sample 20
YCXCFASM
Messages 35
sample 21
YCXCFCMD
Messages 27
Module 6
SEND 11
YCXCFGRP
XCF Group 6
XCF Member 6
YCXCFGRP XCF Group
SEND 10
YCXCFIN1
YCXCFREX 13
YCXCFIN2
YCXCFREX 13
YCXCFMGI
Messages 32
Module 6
Receive RESP 14
Receive SEND 12
SEND 10
YCXCFMGO
Messages 31
Module 7
RESP 13
SEND 11
YCXCFREX 9
YCXCFMRG
sample 19
YCXCFREX
Diagnosis 25
DIAGRMSG 33
Messages 33
Receive SEND 12
REXX 9

Copyright YCOS Yves Colliard Software GmbH



YCXCFIN1 13
YCXCFIN2 13
YCXCFMGO 9
YCXCFWRK 9
YCXCFRMG
Beispiel 9
YCXCFSND
Messages 34
sample 16; 17
YCXCFSTC
Messages 29
Module 6
Receive SEND 12
SEND 10
YCXCFTIM
Messages 28
Module 6
SEND 11
YCXCFTST
sample 19
YCXCFWRK
Messages 30
Module 6
SEND 11
YCXCFREX 9
YXCFA01I
Message 35
YXCFA02I
Message 35
YXCFA03I
Message 35
YXCFA04A
Message 35
YXCFC01A
Message 27
YXCFC02E
Message 27
YXCFC03I
Message 27
YXCFC04I
Message 27
YXCFC05I
Message 27
YXCFC06I
Message 27
YXCFC07I
Message 27
YXCFC08I
Message 27
YXCFC09I
Message 27
YXCFC10I
Message 27
YXCFC11I
Message 27
YXCFC12I
Message 27
YXCFC13I
Message 28
YXCFC14I
Message 28
YXCFD01I
Message 34
YXCFD02A
Message 34
YXCFI02A
Message 32
YXCFI03A
Message 32
YXCFI04A
Message 32
YXCFI05A
Message 32
YXCFI06A
Message 32
YXCFI07A
Message 32
YXCFO01A
Message 31
YXCFO02A
Message 31
YXCFO03A
Message 31
YXCFO04A
Message 31
YXCFO05A
Message 31
YXCFO06A
Message 31
YXCFO07E
Message 31
YXCFO08E
Message 31
YXCFO09A
Message 31
YXCFO10A
Message 31

Copyright YCOS Yves Colliard Software GmbH

YXCFR01I
DIAGRMSG 25
Message 33
YXCFR02A
Message 33
YXCFR03A
Message 33
YXCFR04A
Message 33
YXCFR05A
Message 33
YXCFR06A
Message 33
YXCFR07A
Message 33
YXCFS01I
Message 29
YXCFS02I
Message 29
YXCFS03I
Message 29
YXCFS04I
Message 29
YXCFS05I
Message 29
YXCFS06A
Message 29
YXCFS07A
Message 29
YXCFS08A
Message 29
YXCFS09A
Message 29
YXCFT01A
Message 28
YXCFW01A
Message 30
YXCFW02A
Message 30
YXCFW03A
Message 30

Copyright YCOS Yves Colliard Software GmbH